Covenant Journal of Informatics & Communication Technology. Vol. 10 No. 1, June, 2022

ISSN: ***print*** **2354 – 3566** ***electronics*** **2354 – 3507 DOI: An Open Access**

Journal Available Online

Development of an Enhanced Agility Assessment Model for Legacy Information System

[1] Glory Nosawaru Edegbe, [2]Charles Uche Onianwa (Ph.D)

[1]Department of Computer Science, Edo State Polytechnic, Usen. Benin City.
E-mail: gloryedegbe@gmail.com.
[2]Department of Computer Science, Ambrose Alli University, Ekpoma, Nigeria.
Email: charlesonianwa@aauekpoma.edu.ng

Received: 25.01.2022 Accepted: 31.05.2022
Publication: June, 2022

Abstract- Deciding the moment to end the lifecycle of an information system are often not exhaustively studied. It is essential for an organisation to know when to end the life cycle of their legacy information system when it is no longer able to perform and comply with the changes the organization desires. Prolonging the length of an information system lifecycle could lead to a reduction in software cost. Most of the various metrics presented in literatures on agility measurement, such as Cost, Time, Robustness and Scope of changes (CTRS) and Simplicity, Speed and Scope of changes (3S) and the researchers evaluation methods, e.g., Hierarchy Process (AHP) and Fuzzy Mathematics Analytic are qualitative and usually need to be evaluated by domain experts subjectively. This study therefore developed an enhanced agility assessment model to measure legacy information system quantitatively with the agility factors: Speed, Robustness and Complexity in an educational institution. The adoption of a quantitative metrics methodology will lead to an accurate measurement of the student information system. A stand-alone online assessment system based on agility factors and satisfying the maximum metrics benchmark requirements was used for the model implementation. The results were: Complexity of the largest module=96, Robustness=547.5 hours and Speed 0.5 minutes. The complexity of the module that exceeded 20 can be fixed by reducing the control constructs of the source code modules into submodules, with each not greater than 20. The results obtained indicated that the student information system was still agile. Thus, management should continue with the system.

Keywords: Agility, Software metrics, Legacy Information System, Agility Assessment Model.

1. Introduction

There is no standard definition of a legacy information system (Verbaan, 2010, Crotty & Horrocks, 2017). Aging information systems are usually referred to as legacy (Chen & Rajlich, 2001; Furnweger, Auer & Biffi, 2016); that is an old information system that remains in operation within an organization. Legacy information system is an information system that considerably resists change and evolution (Brodie & Stonebraker, 1998; Bisbal, Lawless, Wu, & Grimson, 1999). Lioyd, Dewar and Pooley (1999) widened the definition of a legacy system to include the business process. Furthermore, Verbaan (2010), sees legacy as a procedural programming paradigm. In this paper, we define legacy information system as an information system that has been in operation for some years and runs on procedural programming language paradigm. On the other hand, the notion of agility is not new; though no rigorous or complete definition of it has been given (Dahmardeh & Pourshahabi, 2011). The word "agility" explains the amount of swiftness and responsiveness of an organization in handling its internal and external happenings (Qumer & Henderson-Sellers, 2006). Agility is the capability of an organization to react speedily and effectively to change (Chandna & Ansari, 2012; Nwokeji, Clark, Barn & Kulkarni, 2015). We see agility as the ability of an organisation to meet or satisfy agility factors' metrics benchmark requirements.

Deciding the moment to end the life cycle of an information system is often not thoroughly researched. The choice to move on to a newer information system is therefore not always satisfactorily justified as the older information system might still be capable of achieving and conforming to the changes the enterprise desires. Extending the length of operation of an information system life cycle can result in cost saving in an application portfolio (Verbaan, 2010). Effort has been made by Wang, Xu, and Zhan (2007), Dahmardeh and Pourshahabi (2011), Chandna and Ansari (2012) to measure legacy information system agility. Nevertheless, some of the metrics offered such as Cost, Time, Robustness and Scope of changes (CTRS) and Simplicity, Speed and Scope of changes (3S) and the researchers evaluation methods, e.g., Fuzzy Mathematics, and Analytic Hierarchy Process (AHP) used in assessing agility are qualitative and in most cases will be evaluated by domain experts subjectively. Thus, imprecise results are often obtained. Although Verbaan's (2010) work was quantitative as software metrics were adopted to implement the agility factors: Quality, Flexibility and Proactivity. He did not include Complexity Speed and Robustness.

The deficiencies in Verbaan's (2010) work include considering Speed as a subset of Flexibility. Strohmaier and Lindstaedt (2011) see flexibility as generally a passive characteristic. Also, Verbaan (2010) adopted MTTF for a repairable software product instead of MTBF which is the metrics used for a repairable product (Speaks, 2005; Chauhan & Pancholi, 2013). We, therefore, enhance the quantitative work of Verbaan (2010) by considering Complexity, Speed and Robustness agility factors and their corresponding software metrics. It is based on the foregoing that we developed an enhanced agility assessment model that will effectively assess legacy information system quantitatively in order to identify and address potential areas in which they would need improvement so that they can increase their ability to change and remain competitive, hence the study. This work is structured as follows: Introduction, Methodology, Results, Related Work and Conclusion (Misra, 2021).

2. Methodology

A developed online assessment system, related and relevant to the software agility factors- Complexity, Robustness and Speed will be used to implement the developed enhanced agility assessment model. The stand-alone online assessment system will be based on the validated relevant software metrics. The input parameters for the assessment system will be from the interrogated case study legacy student information system and the source code. The software metrics was validated by domain experts in the field of Computer Science, Information Systems and Software Engineering to determine their appropriateness. The domain experts were drawn from the academic and the industry. The Metrics for validation were derived from extant literature and rated in Likert format from 1 to 5 based on the relevance of the Metrics to the subject matter. Here, Strongly Disagree (SD) score is 1, Disagree (D) = 2, Undecided (UD) =3, Agree (A) = 4 and Strongly Agree (SA) =5, for each question or statement.

2.1 Data Collection Method

Data used to implement the validated software metrics were collected from the case study student information system that has an available procedural PHP source code and a bug tracking system with historical data regarding bugs, issues and improvement. The student information system has about 240,000 lines of PHP codes with 65 modules. Literature reviews and expert judges or panels were used to measure content validity (Straub, Boudreau & Gefen, 2004). The software metrics selected was validated by five domain experts. A validation threshold for a score of 3.5 was set or adopted. Scoring 3.5 points or higher indicates that the statement was successfully validated.

2.2 Complexity, Robustness and Speed Subsystems

These subsystems enable Complexity, Robustness and Speed Agility factors to be implemented with McCabe cyclomatic complexity metrics, MTBF metrics and Time metrics.

STEPS:

1. Start
2. INTEGER: P, VG, Number_of_Failures
3. REAL: MTBF, SPEED, TIME1, TIME2
4. SPEED=0.0, MTBF=0.0, VG=0
5. Set Benchmark
6. Input P, Number_of_Failures, TIME1, TIME2
7. Compute VG, MTBF, TIME
8. Output
9. Stop

Figure 1: Complexity, Robustness & Speed Subsystem Algorithm

3. Results

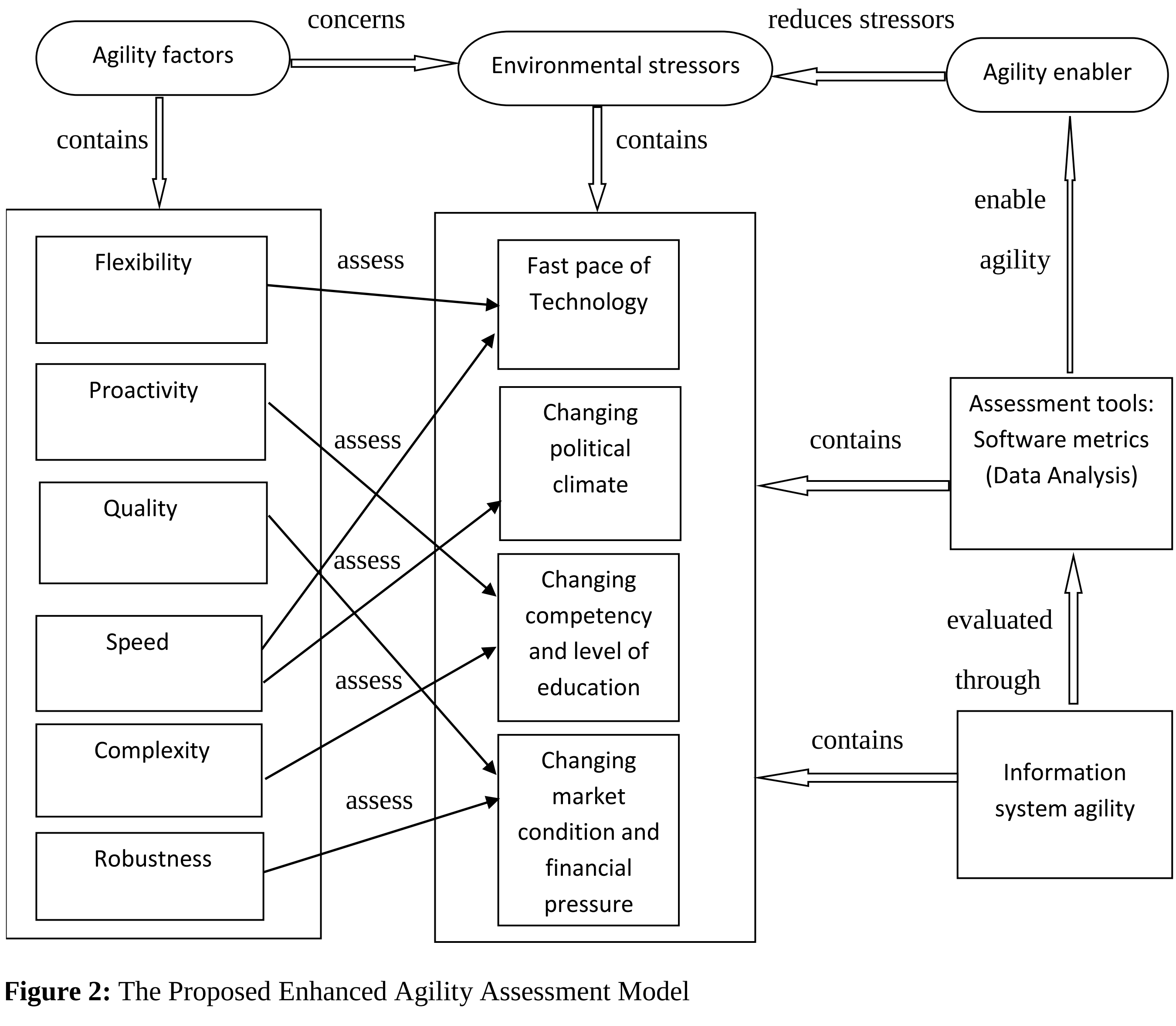

Figure 2: The Proposed Enhanced Agility Assessment Model

There is growing acknowledgement that agility, and information systems are important for the success of modern firms as they face intense rivalry, globalization, and time-to-market pressures (Goldman, Nagel & Preiss 1995; Brown & Eisenhardt, 1997; Sambamurthy, Bharadwaj, & Grover, 2003; Roberts & Grover, 2012; Edegbe, 2015; Felipe, Roldan, & Leal-Rodriguez, 2017; Butt, Misra, Anjum & Hassan (2021). The problem of the existing models is that most of their evaluation methods are subjective. Evaluation methods such as propagation graph and reachability matrix adopted for Component Based Systems (CBS) are somehow complex. Our proposed enhanced agility assessment model evaluation method which make use of software metrics is quantitative. This will, therefore, produce a more accurate result with respect to the agility assessment model features, correctness, reliability and effectiveness. When a legacy information system satisfies the given agility factors metrics benchmark requirements, it is said to be Agile. According to Kurian (2011) being agile means satisfying the given agility factors.

Speed, Robustness and Complexity agility factors with their related metrics, can be used to assess the environmental factors that necessitate the need for organisation agility. Verbaan (2010) framework agility factors: Flexibility, Proactivity and Quality in figure 1 was enhanced with Speed, Robustness and Complexity in the development of our agility assessment model for legacy information system. Verbaan (2010) agility factors metrics are subset of the software metrics adopted in the proposed model implementation. There are some deficiencies with Verbaan (2010) work that needs corrections. He adopted Mean Time Between Failure (MTTF) in his work for software product which is repairable. Mean Time to Failure (MTTF) is used for product that cannot be repaired, unlike MTBF that is used for a repairable product (Speaks, 2005; Chauhan, 2013). Also, the Cyclomatic complexity metrics tool adopted by Verbaan did not identify and display the McCabe cyclomatic complexity count of the source code control constructs. Thus, it will be difficult to know the control constructs that contribute more to the source code complexity in order to refactor the code as the case maybe. Verbaan (2010) equally considered Speed as a subset of flexibility when it is not. Strohmaier and Lindstaedt (2011), perceived flexibility as a commonly passive characteristic. Consequently, our enhanced agility assessment model is presented in figure 2. Some of the environmental issues are: Fast pace of technology development. New technologies roll out is on a regular basis. There are situations where an information system of two to three years in operation is already a legacy system due to the fast pace of technological innovation. Changing political climate can affect the survival of an information system as new government in most cases comes up with new policies which results in environmental instability. Changing competency and level of education and changing market condition and financial pressure are also issues that need to be confronted. Organisations need to constantly determine the level of staff competencies in the ever dynamic environment where information systems are becoming more complex and challenging. Also, there is high technical staff turnover in the information technology (IT) sector. In view of this, there is need for the regular training of technical staff to enable them get abreast of current development and trends in the IT industry. There is also the need to constantly measure staff level of competence in order to fill the skill gap either from within the organisation or outside the organisation. For organisation to survive, they have to consistently meet up with changing consumer taste. Otherwise, there might be a lot of financial pressure on the organisation.

Implementing our developed agility assessment model in figure 2, a student information system was used as our case study. The metrics benchmark adopted is in table 1 were derived from literature and top management of the case study student information system.

Table 1: Adopted Benchmark for Student Information System Implementation

S/N	Metrics	Benchmark	Authors
1.	Cyclomatic Complexity	20	Mendez and Tinetti (2014).
2.	Time	3 Minutes	Management.
3.	MTBF	148 Hours	Verbaan (2010).

The input data for Robustness and Speed agility factors were derived from querying the students' information system, while that of Complexity was derived from the student information system PHP application program source code. The data were entered into the developed online assessment system to generate the program output in table 2.

Table 2: Cyclomatic Complexity(CC), MTBF and Time Metrics Results

Agility Factors	Metrics	Result
1. Complexity	CC (P+1)	96
2. Robustness	MTBF (Hrs)	547.5
3. Speed	Time (Minute)	0.5

Source: Author (2020)

Evaluating the source code with the online assessment system complexity subsystem, some of the PHP code control constructs were more than 20. A module of 159 lines of code had control constructs of 12 with Cyclomatic complexity of 13. Also, a module of 201 lines of code had a control constructs of 5 with Cyclomatic complexity of 6. Another module of 395 lines of code had control constructs of 95 with Cyclomatic complexity of 96. For Robustness, 16 software failures were recorded in a year. Entering the numbers of software failures into the Robustness subsystem, MTBF=547.5 hours. Time was considered to measure Speed at two different intervals (Time1 & Time2) of 6 months. We used same computer on the same network and took into account peak times. Here, 20MB data size of students' sessional result reporting sheets of 10,000 records were generated in Excel format. Using the Speed subsystem, the generation time for Time1 = 4 minutes and that of Time2= 4.5 minutes. The Time differential=0.5 minutes.

Comparing the result in table 3 with the adopted benchmark metrics in table 2 indicate that the Robustness and Speed of the student information system are okay, as they fall within the metrics threshold values. But, the PHP source code is complex because the cyclomatic complexity of the source code module of 96 exceeds the benchmark value of 20. Thus, the module needs redesign. This can be achieved by reducing the control constructs of the source code modules having cyclomatic complexity greater than 20 and also subdividing each module into many modules and submodules to keep the size of the modules controllable and permit the testing of all independent paths as well as detect and debug faults from the source code. We can conclude that the student information system is still agile as it satisfied the accepted agility factors' metrics benchmark requirements.

Adopting the proposed model will enable us to determine when to end the lifecycle of a legacy system (when it can no longer perform and comply with the changes the enterprise desires). Extending the length of an information system lifecycle could result in cost reduction in an application portfolio.

4. Related Work

Verbaan (2010) proposed a technique of assessing legacy information system. He listed likely areas in which a legacy information system would require enhancement in order to increase its ability to change. He studied different category of metrics by choosing flexibility, quality and pro-activity as agility factors with the matching metrics. Also, Speed was not selected as they believe it is a subset of Flexibility. Strohmaier and Lindstaedt (2011) perceived flexibility to be a rather commonly passive characteristic. Verbaan (2010) adopted MTTF for a repairable software product instead of MTBF which is the metrics used for a repairable product (Speaks, 2005; Chauhan & Pancholi, 2013).

Edegbe, Chiemeke and Ihama (2013) in their paper titled "Overview of Software System Agility Assessment Models" extends the agility factors proposed by Verbaan (2010) to include Innovation. However, legacy information system is not innovative (Verbaan, 2010).

Bakar, Razali and Ismail (2018) legacy information system assessment model focused on the public sector. Their research adopted qualitative approach which incorporates the theoretical and empirical phase. The theoretic part was conducted by examining existing literature of the information system assessment models and methods which include ISO 25010, ISO 25012, Hierarchical Model and Renaissance Method. The empirical approach was conducted by employing interviews with informants involved in the use of legacy information system. Data from the theoretical and empirical study were analysed using content analysis.

Qumer and Henderson-Sellers (2006) stated that agility can be measured in terms of the five variables or (features) namely: a. Flexibility (FY), b. Responsiveness (RS), c. Leanness (LS), d. Learning (LG) and e. Speed (SD). They developed a mathematical model for the implementation.

Rathor and Batra (2016) developed a structural model to assess the effects of delivery capability and agility on software development success and also quantify the effects of antecedent process variables on delivery capability and agility. To test the research model and hypotheses, survey data was collected using an online questionnaire from IT professionals that have adopted agile methodology for software development. For the evaluation of the survey data, Partial least squares (PLS) was used. Omorodion and Damasevicius (2020) proposed that agile software developments methods and their corresponding software metrics which are quantitative can be used to assess the quality growth of a product being developed.

Jin et al. (2007) work on legacy information system was on database-centric information systems which often process a large amount of data with stringent performance requirements. Their area of focus was on performance evaluation and prediction for legacy information systems when they are subjected to an upsurge in workload and database loading. Their method combines the use of benchmarking, production system monitoring and performance modelling.

Gandomani and Nafchi (2014) presented a model to measure agility degree of agile software companies. The proposed model can compute the Agility of an organisation based on the adopted practices in that organisation. This study identified the importance of agile practices in being Agile. The foundations of the proposed model are agile practices and their importance in achieving agile values. Their work focused on software development Process.

Jassbi et al. (2010) developed approach was based on Adaptive Neuro Fuzzy Inference System (ANFIS) for evaluating agility in supply chain. They considered agility capabilities such as Flexibility, Competency, Cost, Responsiveness and Quickness. Absence of efficient measuring tool for agility of supply chain system motivated them to develop a technique with above-mentioned functionality. The vague nature of the associated concepts qualities convinced them to adopt fuzzy concepts and combined this powerful tool with Artificial Neural Network concepts in favour of gaining ANFIS as an effective and efficient device for development and surveying of their unique procedure.

Kumar et al. (2016) work commenced with the development of a Supply Chain Agility Assessment Model in a manufacturing organisation. The model is comprehensive in nature as it includes Five Enablers and Twenty-Two different Agile Supply Chain Criteria and various Agile Supply Chain Attributes. The Fuzzy logic method was used to assess the Supply Chain Agility. The output of the Project included the Supply Chain Agility Index, Fuzzy Performance Important Index of Various Agile Supply Chain Attributes and Identification of Principal Obstacles. The improvements for supply chain agility improvement were derived from within the company. The implementations of the results led to enhanced profitability and increase in customer domain of the organization.

Nazir and Pinsonneault (2012) work investigated the link between IT and firm agility through an electronic integration perspective. The electronic integration perspective framework suggests that IT applications affect the sensing and responding components of agility through internal and external integration. The framework in addition describes the mediating roles of knowledge exploitation, knowledge exploration and process coupling.
Chandna and Ansari (2012) recommended fuzzy inference systems (FIS) that is design in many steps: fuzzification, aggregation of antecedents, inferencing, composition, and defuzzification for measuring agility. The flaws of this fuzzy logic method are that the membership functions of linguistic variables hinge on the managerial perception of the decision-maker. Shahrabi (2011) work also used fuzzy logic as a tool to evaluate their agility assessment model called Grason model in order to determine the agility of an organization and its relationships.

Avazpour et al. (2014) proposed framework was based on the fuzzy multiple criteria decision making method to identify the most suitable agility enablers to be applied by companies. Using fuzzy logic to address the ambiguity in agility evaluation, the fuzzy Prioritization Method was applied to determine weights of the agility attributes as their criteria. In order to rank the agility enablers Similarity-Based Approach was adopted as their alternatives. The framework was implemented in a real case involving a subsidiary company of the National Iranian Gas Company. The proposed framework helps the company to concentrate on the most effective enablers and develop strategies to implement them based on their priority.

Wang et al. (2007) stated that some of the various metrics presented in literatures such as Cost, Time, Robustness and Scope of changes (CTRS) and Simplicity, Speed and Scope of changes (3S) and the researchers evaluation methods, e.g., Analytic Hierarchy Process (AHP) and Fuzzy Mathematics used to integrate these metrics together to get the final results are qualitative and usually need to be assessed by domain experts subjectively. The results that will be derived will not be accurate. To solve these deficiencies above, Wang et al. (2007) presented an agility evaluation method called propagation graph and reachability matrix to precisely measure agility for CBS. The limitation is that the method is somehow complex.

Aggoune (2012) et al extends the concept of agility to the e-government field through an evaluative framework for the measurement of e-government information system (E-GIS) agility. The key idea of this framework is to combine the fundamental parts of E-GIS with their operational parameters to evaluate the overall agility of the system. One of the benefits of this practical framework is that agility parameters are assessed with the help of a quantitative metrics, which allow decision-makers to inspect and compare different systems at different agility levels. The evaluated framework presented is only theoretical.

Imache et al. (2012) extended the work of Izza et al. (2008). They proposed a fuzzy logic based assessment methods in order to assess, regulate and preserve continuously the Information System agility. They also propose a prototype implementation and an application of the proposed approach to a tour operator enterprise. Their use of the POIRE framework was based on two main principles: Urbanization and continuous improvement. The limitation of the presented model is that it neglects the mutual interaction between the different dimensions' factors and criteria of the enterprise information system, so the obtained results of the agility are not essentially the best one. Also, their work mainly focuses on agile information system rather than the technological perspective.

Trabelsi and Abid (2013) work was still on Urbanization. They proposed urbanization framework that aims to simplify information system by improving communication between its components and to ensure its evolution. In an exploratory approach, the study examines the state of urbanization of information system in Tunisian company and verifies the agility of urbanized information system (UIS). This was done in order to ascertain the evolution of information system and guarantee the agility facing the environment turbulence. However, this study has several weaknesses. In the first instance the sample size of the study was limited to only private firms in Tunisia. The generalizability of the results may be limited because of the small sample size when compared to the overall population, including the Small and Medium Enterprises (SME) in other study. Second, the number of variables used in this study is also limited to agility, interoperability and flexibility.

5. Conclusion

We have developed an enhanced agility assessment model to measure legacy information system agility using agility factors: Speed, Robustness and Complexity. The agility factors were evaluated with a developed online assessment system using software metrics relevant to the procedural PHP programming language case study of student information system running mainly a procedural PHP programming language application. The result of the implementation indicates that the PHP source code module with the value of 96 is complex and needs to be redesigned. The source code modules greater than 20 should be subdivided into modules not greater than 20. The results of Speed and Robustness were satisfactory. The online assessment system for the software metrics was written in PHP programming language, JavaScript, HTML, CSS and MySQL. PHP programming language was adopted because it is an open source with advanced features, it supports MySQL and can create dynamic web pages. Our major findings indicate that the student information system is still agile as it satisfied agility factors' metrics benchmark requirements.

REFERENCES

Aggoune S. Imache R., Mezghiche M. (2012). Evaluation Framework for the Measurement of E-government Information System Agility. International Journal of Digital Society (IJDS), 3(4).

Avazpour R. Ebrahimi E, Fathi, M.R. (2014). Prioritizing Agility Enablers Based on Agility Attributes using Fuzzy Prioritization Method and Similarity-Based Approach. International Journal of Economy, Management and Social Sciences, 3(1), 143-153.

Bakar A.K.A, Razali R, Ismail F.F. (2018). The Assessment Model for Legacy Information System for Public Sector. Journal of Theoritical and Applied Information Technology, 96(17).

Bisbal J, Lawless D, Wu B, Grimson J. (1999). Legacy Information Systems: Issues and Directions. IEEE Software, 16(5), 103-111.

Brodie M. L, Stonebraker M. (1995). Migrating Legacy Systems: Gateways, Interfaces and the Incremental Approach. San Francisco, CA: Morgan Kaufmann.

Brown S. L, Eisenhardt K. M. (1997). The Art of Continuous Change: Linking Complexity Theory and Time-Paced Evolution in Relentlessly Shifting Organizations. Administrative Science Quarterly, 42(1), 1-34.

Butt S. A, Misra S, Anjum M. W, Hassan S. A. (2021). Agile Project Development Issues during COVID-19. In International Conference on Lean and Agile Software Development (pp. 59-70). Springer, Cham.

Chandna R, Ansari S. R. (2012). Fuzzy Approach to Measure Manufacturing Agility along with Developing Fuzzy Inference Systems in MATLAB. International Journal of Engineering and Technology 6(2).

Chauhan N. D, Pancholi N. H. (2013). Guidelines to Understanding to Estimate MTBF. International Journal for Scientific Research & Development| 1(3)

Chen K, Rajlich V. (2001). RIPPLES: Tool for Change in Legacy Software. Proceedings of ICSM ’01: IEEE Computer Society Press (pp. 230-239). Los Alamitos: CA.

Crotty J, Horrocks I. (2017). Managing Legacy System Cost: A Case Study of a Meta-Assessment Model to Identify Solutions in a Large Financial Services Company. Applied Computing and Informatics, 13,175-183.

Dahmardeh N, Pourshahabi V. (2011). Agility Evaluation in Public Sector using Fuzzy Logic. Iranian Journal of Fuzzy Systems 8 (3), 95-111.

Edegbe G. N. (2015). A Review of Agility Factors (Metrics) for Measuring Information systems, International Journal of Advances in Electronics and Computer Science, 2(9), 62-64.

Edegbe G. N, Chiemeke S.C, Ihama E.I. (2013) Overview of Software System Agility Assessment Models, AAU Journal of Management Sciences 4(2), 117-131.

Felipe C. M, Roldan J. L, Leal-Rodriguez A. L. (2017). Impact of Organizational Culture Values on Organizational Agility. Sustainability, 9 (12).

Furnweger A, Auer M, Biffi S. (2016). Software evolution of legacy systems: A Case Study of Soft-Migration. Proceedings of the 18th International Conference on Enterprise Information Systems (ICEIS 2016), Volume 1, (pg. 413-424).

Gandomani T. J, Nafchi, M. Z. (2014). Agility Assessment Model to Measure Agility Degree of Agile Software Companies. Indian Journal of Science and Technology, 7(7), 955–959.

Goldman S. L, Nagel R. N, Preiss K. (1995). Agile Competitors and Virtual Organizations: Strategies for Enriching the Customer. New York, NY: Van Nosstrand Reinhold.

Imache R, Izzah S, Ahmed-Nacer M. (2012). An Enterprise Information System Agility Assessment Model. ComSIS, 9(1).

Izza S, Imache R, Vincent L, Lounis Y. (2008). Interoperability for Enterprise Software and Applications. In K, Mertins, R, Ruggaber, K, Popplewell, X, Xu. (Eds.), An *Approach* for the Evaluation of the Agility in the Context of Enterprise Interoperability. Interoperability for Enterprise Software and Applications Conference (I-ESA‟08) Springer, London, 3-14.

Jassbi J, Seyedhosseini S.M, Pilevari N. (2010). An Adaptive Neuro Fuzzy Inference System for Supply Chain Agility Evaluation. International Journal of Industrial Engineering & Production Research, 20(4), 187-196.

Jin Y, Tang A, Han J, Liu Y. (2007*).* Performance Evaluation and Prediction for Legacy Information System. 29th International conference on Software Engineering (ICSE'07) IEEE.

Kumar C.S, Kumar V. K, Kumar D. (2016). A Scientific Approach to the Design and Development of a Model to Improve the Assessment Level by Fuzzy Logic. European Journal of Engineering and Technology, 4 (3).

Kurian T. (2011). A Fuzzy Based Approach for Estimating Agility of an Embedded Software Process. Avionics Systems Business.

Lloyd A.D, Dewar R, Pooley R. (1999). Legacy Information Systems and Business Process Change: A Patterns Perspective. Communications of the Association for Information Systems, 2(24).

Mendez M, Tinetti F. G. (2014). Integrating Software Metrics for Fortran Legacy into an IDE. XX Argentine Congress of Computer Science (Buenos Aires, 2014).

Misra S. (2021). A Step by Step Guide for Choosing Project Topics and Writing Research Papers in ICT Related Disciplines, Communications in Computer and

Information Science, Volume： 1350 Page：727-744.
Misra S, Omorodion F. M, Damasevicius R. (2020). Metrics for Measuring Progress and Productivity in Agile Software Development. In International Conference on Innovations in Bio-Inspired Computing and Applications (pp. 469-478). Springer, Cham
Nazir S, Pinsonneault A. (2012). IT and Firm Agility: An Electronic Integration Perspective. Journal of Association of Information Systems, 13(3), 150-171.
Nwokeji J. C, Clark T, Barn B, Kulkarni V. (2015). A Conceptual Framework for Enterprise Agility. Proceedings of the 30th Annual ACM Symposium on Applied Computing. Salamanca, Spain, 1249-1250.
Qumer A, Henderson-Sellers B. (2006). Measuring Agility and Adoptability of Agile Methods: A 4- Dimensional Analytical Tool. IADIS International Conference Applied Computing 2006, IADIS Press, 503-507.
Rathor S, Batra D. (2016). Tradeoffs Between Delivery Capability and Agility in Software Development. Proceedings of the 15th AIS SIGSAND Symposium, Lubbock, TX, May 12-14.
Trabelsi L, Abid I. M. (2013). Urbanization of Information Systems as a Trigger for Enhancing Agility: A State in the Tunisian Firms. European Journal of Business and Management, 5(5).
Sambamurthy V, Bharadwaj A, Grover, V. (2003). Shaping Agility Through Digital Options: Reconceptualising the role of information technology in contemporary firms. MIS Quarterly, 27 (2), 237-263.
Shahrabi B. (2011). The Agility Assessment Using Fuzzy Logic. World Applied Sciences Journal, 13 (5), *1112-1119.*
Speaks S. (2005). Reliability and MTBF Overview. Vicor Reliability Engineering, Andover.
Straub D, Boudreau M, Geden D. (2004). Validation of guidelines for IS Positive Research. Communications of the Association for Information Systems, 13(1), 380-427.
Strohmaier M, Lindstaedt S. N. (2011). Beyond Flexible Information Systems: Why Business Agility Matters. Austria: Know-Center Graz, Inffeldgasse 21a, 8010 Graz.
Verbaan M. (2010). Legacy Information Systems, Can They Be Agile? A Framework for Assessing Agility. Communications of the IIMA, 10 (4), 1-14.
Wang Z, Xu X, Zhan D. (2007). Agility Evaluation of Component-Based Systems. Journal of Information Science and Engineering, 23, 1769-1783.